

REMARKS

Applicants respectfully request further examination and reconsideration in view of the arguments set forth fully below. Claims 1-26 were previously pending in this Application. Within the previous Office Action, Claims 1-26 have been rejected. By the above amendment, new Claim 27 has been added. Accordingly, Claims 1-27 are now pending in the application.

Rejections Under 35 U.S.C. § 102

Within the previous Office Action, Claims 1-3, 5-16 and 21-26 have been rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent Application Publication No. 2002/0013852 to Janik (hereinafter "Janik"). The Applicants respectfully disagree. Janik does not teach *prefetching* audio/visual content based on a preference corresponding to a user.

Janik teaches a system for providing content, management, and interactivity for thin client devices. Janik teaches a capability for determining and aggregating the content objects presented to a specific user on content selection web pages which is derived from content preference selections provided by the user. [Janik, ¶ 0082] Janik further teaches time-based automation of the accessing, caching and streaming of content from the Internet at times prescribed by the user or at times derived by direction given by the user through the GUI content editors. [Janik, ¶ 0105] Janik does not teach *prefetching* audio/visual content based on a preference corresponding to a user.

Within the Response to Arguments section of the previous Office Action, it is argued that Janik teaches prefetching a content item based on a prefetch parameter and a current display window. However, no cite or support is given for this proposition. Janik teaches that there are three functional modes including setup, real-time user controlled content/data delivery and automatic content/data delivery. [Janik, ¶ 0129] Janik further teaches that

[i]n automatic mode, content 10 that the user has selected for playback in the content editor is sent automatically to the playback device, based on some prescribed time setting that was pre-set by the user. A scheduling function in core module 42 compares time inputs listed in system control application database 96 with the current state of PC 34 system timer. When a match occurs between a time input in system control application database 96 and the current state of PC 34 system timer, core module 42 initiates the delivery of content 10 to client device 78. [Janik, ¶ 0165]

Accordingly, Janik teaches automatically obtaining content based on a time schedule. Janik does not teach *prefetching* audio/visual content based on a preference corresponding to a user.

In contrast to the teachings of Janik, the methods and apparatus described within the present application, organize audio/visual content and *prefetch* selected audio/visual content configured to be displayed to a user. A presentation layer takes into account the preferences and use patterns of a user. [Present Specification, page 8, lines 17-20] In one embodiment, audio/visual content is pre-sorted according to the use patterns of the user. [Present Specification, page 8, lines 20-21] In another embodiment, the audio/visual content is *pre-fetched* according to the use patterns of the user. [Present Specification, page 8, lines 21-22] As also taught within the Present Specification,

[t]o minimize the wait time of utilizing the selected audio/visual content, the method and apparatus for presenting content *prefetches* the audio/visual content based, in part, on the location of the particular audio/visual content listing relative to the listings displayed in current display window 635. For example, if the methods and apparatuses for presenting content are configured to *prefetch* audio/visual content within the current display window 635, then the audio/visual content represented within the display windows 625, 630, 640 and 645 are prefetched. As the user scrolls through the various display windows, the *prefetched* audio/visual content dynamically changes. In one illustrative employment, content items that correspond to each content item listing displayed in $n \pm 1$ windows are *prefetched*, where $n=1$. In some embodiments, the *prefetched* content represented in the display windows 625, 630, 640 and 645 are temporarily stored within the prefetched buffer 315. [Present Specification, page 17, lines 5-17, emphasis added]

As described above, Janik teaches time-based automation of the accessing, caching and streaming of content at times prescribed by the user or at times derived by direction given by the user. As further described above, Janik does not teach *prefetching* audio/visual content based on a preference corresponding to a user. Janik teaches automatically obtaining content based on a time schedule.

The independent Claim 1 is directed to a method comprising identifying a preference corresponding to a user, detecting a current display window and *prefetching* at least one audio/visual content in response to the current display window and the preference. As described above, Janik teaches time-based automation of the accessing, caching and streaming of content at times prescribed by the user or at times derived by direction given by the user. As further described above, Janik does not teach *prefetching* audio/visual content based on a preference corresponding to a user. Janik teaches automatically obtaining content based on a time schedule. For at least these reasons, the independent Claim 1 is allowable over the teachings of Janik.

Claims 2, 3 and 5-11 are all dependent on the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Janik. Accordingly, Claims 2, 3 and 5-11 are all also allowable as being dependent on an allowable base claim.

The independent Claim 12 is directed to a system comprising means for identifying a preference, means for organizing audio/visual content using a parameter, means for detecting a current display window and means for *prefetching* at least one audio/visual content in response to the current display window and the preference. As described above, Janik teaches time-based automation of the accessing, caching and streaming of content at times prescribed by the user or at times derived by direction given by the user. As further described above, Janik does not teach means for *prefetching* audio/visual content based on a preference corresponding to a user. Janik teaches automatically obtaining content based on a time schedule. For at least these reasons, the independent Claim 12 is allowable over the teachings of Janik.

The independent Claim 13 is directed to a method comprising detecting an activity, setting a prefetch parameter based on the detected activity, detecting a current display window and *prefetching* a content item based on the prefetch parameter and the current display window. As described above, Janik teaches time-based automation of the accessing, caching and streaming of content at times prescribed by the user or at times derived by direction given by the user. As further described above, Janik does not teach *prefetching* a content item based on a prefetch parameter and a current display window. Janik teaches automatically obtaining content based on a time schedule. For at least these reasons, the independent Claim 13 is allowable over the teachings of Janik.

Claims 14-16, 21 and 22 are all dependent on the independent Claim 13. As described above, the independent Claim 13 is allowable over the teachings of Janik. Accordingly, Claims 14-16, 21 and 22 are all also allowable as being dependent on an allowable base claim.

The independent Claim 23 is directed to a system comprising a media container configured for storing an audio/visual content item, a prefetch buffer configured for temporarily storing a *prefetched* audio/visual content item and a presentation layer configured for transmitting the *prefetched* audio/visual content item to the prefetch buffer based on a user's preference and a current display window. As described above, Janik teaches time-based automation of the accessing, caching and streaming of content at times prescribed by the user or at times derived by direction given by the user. As further described above, Janik does not teach *prefetching* audio/visual content based on a user's preference and a current display window. Janik teaches automatically obtaining content based on a time schedule. For at least these reasons, the independent Claim 23 is allowable over the teachings of Janik.

Claims 24-26 are all dependent on the independent Claim 23. As described above, the independent Claim 23 is allowable over the teachings of Janik. Accordingly, Claims 24-26 are all also allowable as being dependent on an allowable base claim.

Rejections Under 35 U.S.C. § 103

Within the Office Action, Claims 4 and 17-20 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Janik. The Applicants respectfully disagree. As the basis of this rejection, Official Notice has been taken that it is old and well known in the art for a user's preference information to be stored and obtained by identifying the user through a variety of methods, for example, the use of a username and a password. The Applicants respectfully disagree with this conclusion and the taking of Official Notice to support this rejection.

Claim 4 is dependent on the independent Claim 1. Claims 17-20 are dependent on the independent Claim 13. As described above, the independent Claims 1 and 13 are allowable over the teachings of Janik. Accordingly, Claims 4 and 17-20 are all also allowable as being dependent on an allowable base claim.

For the reasons given above, the applicant respectfully submits that the claims are now in a condition for allowance, and allowance at an early date would be appreciated. Should the Examiner have any questions or comments, they are encouraged to call the undersigned at (408) 530-9700 to discuss the same so that any outstanding issues can be expeditiously resolved.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: November 26, 2007

By: Jonathan O. Owens

Jonathan O. Owens
Reg. No. 37,902
Attorneys for Applicant(s)